NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

SEA-SHORE INTERFACE ROBOTIC DESIGN

by

Timothy L. Bell

June 2014

Thesis Advisor: Richard Harkins

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
06-20-2014	Master's Thesis	01-01-2013 to 06-20-2014

4. TITLE AND SUBTITLE

SEA-SHORE INTERFACE ROBOTIC DESIGN

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Timothy L. Bell

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Postgraduate School
Monterey, CA 93943

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

N/A

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited

13. SUPPLEMENTARY NOTES

The views expressed in this document are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol Number: N/A.

14. ABSTRACT

An exoskeleton platform was developed, prototyped and tested for mobility performance in a beachfront environment. New platform, drive-train, motor-controller and wheel design were employed in the experiment. The objective was to improve on the shortcoming of previous NPS research. Three wheel-designs were tested during fixed pattern tests on grass, concrete and sand. Data suggests that, with regard to power consumption, there is a marginal difference on preferred wheel design. The sparse print round wheel showed promise in heavy vegetation; however, the Wheg^{TM} wheel proved to be the most versatile on various terrains. This suggests that a Wheg^{TM} wheel with improved round wheel characteristics would be optimal for various beachfront terrains.

15. SUBJECT TERMS

Robotics, Robot, Amphibious Vehicles, Mobility, Surf-Zone, Autonomous, Wheg, exoskeleton

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER *(include area code)*
Unclassified	Unclassified	Unclassified	UU	79	

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

SEA-SHORE INTERFACE ROBOTIC DESIGN

Timothy L. Bell
Lieutenant, United States Navy
B.S., University of South Florida, 2007

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN APPLIED PHYSICS

from the

NAVAL POSTGRADUATE SCHOOL
June 2014

Author: Timothy L. Bell

Approved by: Richard Harkins
Thesis Advisor

Andres Larraza
Chair, Department of Physics

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

An exoskeleton platform was developed, prototyped and tested for mobility performance in a beachfront environment. New platform, drive-train, motor-controller and wheel design were employed in the experiment. The objective was to improve on the shortcoming of previous NPS research. Three wheel-designs were tested during fixed pattern tests on grass, concrete and sand. Data suggests that, with regard to power consumption, there is a marginal difference on preferred wheel design. The sparse print round wheel showed promise in heavy vegetation; however, the Wheg^{TM} wheel proved to be the most versatile on various terrains. This suggests that a Wheg^{TM} wheel with improved round wheel characteristics would be optimal for various beachfront terrains.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

1 Introduction **1**
1.1 Background 2
1.2 Objectives 2

2 Robot Design **5**
2.1 Dynamics and Control 5
2.2 Design 6

3 Experiments and Results **15**
3.1 Data Collection 15
3.2 Data Reduction and Calculations 17
3.3 Experimental Tests and Results 20
3.4 Current Use Summary 23

4 Conclusion **27**
4.1 Platform Capability 27

Appendices

A Basic Autonomous Pattern Run **29**

B Filtered Current Data **41**

C Raw Current Data **49**

List of References **57**

Initial Distribution List **59**

THIS PAGE INTENTIONALLY LEFT BLANK

List of Figures

Figure 1.1 RHex is a biologically inspired robot by Boston Dynamics, from [1] 1

Figure 1.2 LT Bell's Wheg™ design. 2

Figure 2.1 DARc's components are modular and can be easily replaced or alternate components can be used. 6

Figure 2.2 The base is designed to receive a Pelican™ 1400 case. It allows the electronics module and the drive assembly to be easily changed in a modular fashion. 7

Figure 2.3 The square aluminum tube houses the drive assembly. The motor is mounted to the outside of the tube. Each transmission is independent of the other. 8

Figure 2.4 The chassis is formed of three separate components, the two transmissions and the base, allowing for easy system modification. . . 8

Figure 2.5 Brushed permanent magnet DC motor, variable speed and reversible, 24VDC, Reduction Ratio: 1:49, Rated Torque: 16 kgf-cm, Rated Speed: 122 RPM, and no Load Current: < 500mA, from [7]. . . . 9

Figure 2.6 The motors were installed in the forward portion of the robot and were placed in a protected area. The rubber sleeve acts as a waterproof covering. 9

Figure 2.7 The Pelican 1400 case houses the electronic module. 10

Figure 2.8 The interior has ample room for the controller, batteries and RC module. 10

Figure 2.9 The Roboteq™ HDC2450 is a Brushed DC Motor Controller, Dual Channel, 150A, 50V, Encoder in, USB, CAN capable with on-board 32-bit microcomputer and uses Micro Basic as the language, from [8]. 11

Figure 2.10 This figure depicts some of the additional capability of the Roboteq controller, from [9]. 11

Figure 2.11 25.4 cm diameter by 2.54 cm thick five leg Wheg with automotive timing belt as tread designed by LT Bell. 12

Figure 2.12 This Wheg was designed with paddle elements to move through sandy and muddy environments while avoiding collection of debris. 12

Figure 2.13 25.4 cm diameter by 2.54 cm thick round wheel with automotive timing belt as tread. 13

Figure 2.14 A commercial rubber nylon tube type 4.10/3.50-4 tire to test how a wider tire operates in the surf-zone environment. 14

Figure 2.15 Depicted is the previous Surf-zone robot, MONTe, which illustrates an excellent example of a retractable tail for righting and climbing situations, from [3]. 14

Figure 3.1 Experimental Route Plan. 16

Figure 3.2 Current data from the Roboteq is sent wirelessly to a laptop and processed by MATLAB to get desired graphs. 16

Figure 3.3 The Proxim^{TM} modems are capable of 11540 baud and can use serial or Ethernet connections. 17

Figure 3.4 Raw Current Data For Round Wheel in Grass. 18

Figure 3.5 Raw Current Data with Filtered Curve Overlayed 19

Figure 3.6 The data has been processed and now has a smooth curve. 20

Figure 3.7 Current Use Summary by Surface Type. 24

Figure B.1 Round Wheel Current on Grass. 41

Figure B.2 Rubber Wheel Current on Grass. 42

Figure B.3 Wheg Current on Grass. 42

Figure B.4 Round Wheel Current on Concrete. 43

Figure B.5 Rubber Wheel Current on Concrete. 43

Figure B.6 Wheg Current on Concrete. 44

Figure B.7 Round Wheel Current on Sand. 44
Figure B.8 Rubber Wheel Current on Sand. 45
Figure B.9 Wheg Current on Sand. 45
Figure B.10 Rubber Wheel Current on Grass Hill. 46
Figure B.11 Rubber Wheel Current on Grass Hill. 46
Figure B.12 Wheg Current on Grass Hill. 47

Figure C.1 Round Wheel Current on Grass Raw Data. 49
Figure C.2 Rubber Wheel Current on Grass. 50
Figure C.3 Wheg Current on Grass. 50
Figure C.4 Raw Round Wheel Current on Concrete. 51
Figure C.5 Raw Rubber Wheel Current on Concrete. 51
Figure C.6 Raw Wheg Current on Concrete. 52
Figure C.7 Wheg Current on Concrete. 52
Figure C.8 Raw Round Wheel Current on Sand. 53
Figure C.9 Raw Rubber Wheel Current on Sand. 53
Figure C.10 Raw Wheg Current on Sand. 54
Figure C.11 Raw Round Wheel Current on Grass Hill. 54
Figure C.12 Raw Rubber Wheel Current on Grass Hill. 55
Figure C.13 Raw Wheg Current on Grass Hill. 55

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

Table 3.1	Peak Current, Average Current and Power for Wheels on Grass.	21
Table 3.2	Peak Current, Average Current and Power for Wheels on Concrete.	21
Table 3.3	Peak Current, Average Current and Power for Wheels on Sand.	22
Table 3.4	Peak Current, Average Current and Power for Wheels on Hill.	23

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

NPS Naval Postgraduate School
UAV Unmanned Aerial Vehicles
UUV Unmanned Underwater Vehicles
ALR Autonomous Land Rovers
RC remote control
VDC voltage direct current
CAD Computer Aided Design
RPM revolutions per minute
PID proportional-integral-derivative

THIS PAGE INTENTIONALLY LEFT BLANK

Acknowledgements

First, I would like to thank my wife for all her patience in dealing with a student earning a physics master's degree. Her support and encouragement were vital in allowing me to finish my thesis and degree.

To my thesis advisor, thank you for encouraging me to keep up with the physics program. I truly feel accomplished.

To Steve Jacobs, thanks for building an awesome robot with me. I certainly could not have done this without your knowledge and expertise.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1:
Introduction

It is apparent that the role of autonomous vehicles is increasingly important in all aspects of modern society. The obvious military use is in the operational theater. Additionally, there are many uses in the civilian sector such as search, rescue and recovery.

Unmanned Aerial Vehicles (UAV), Unmanned Underwater Vehicles (UUV) and Unmanned Land Rovers (ULR) are well understood and there are many acceptable production models for military and civilian applications. However, there is not a surf-zone style robot in the operational stage. One of the leaders in the field is Boston Dynamics, they have not focused on the surf-zone, however.

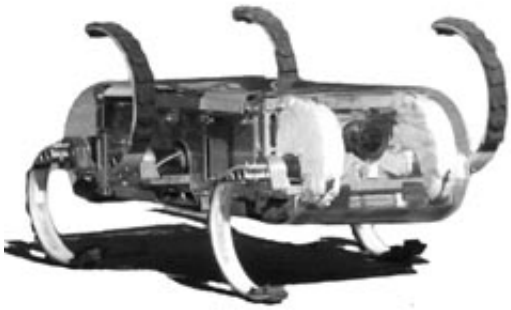


Figure 1.1: RHex is a biologically inspired robot by Boston Dynamics, from [1]

The challenge is for autonomous vehicles to make the transition from water to land through the tumultuous wave zone, with the ability to be self-righting and to navigate various terrain states.

The goal for the Surf-zone robot is that it be portable and low-cost, small enough for Special Forces personnel to carry, and sufficiently cost effective to be disposable. The platform will need to provide an excellent capability to map an area for a beach landing or to gather sources of intelligence, such as video or sound.

1.1 Background

Prior models of the Surf-zone robot include a Foster Miller Lemming tracked platform [2], which demonstrated semi-autonomous behavior and moderate mobility. AGBOT [3] was excellent at climbing, but was not waterproof and tended to rattle components loose because of its lack of a suspension. MONTe included a complex suspension system that proved very successful. It was fabricated out of 3-D print material, but proved to be too fragile [4].

The later of these models, AGBOT and MONTe, used the Wheg^{TM} design. The Wheg uses a round construct with protrusions; the name is a contraction of the words wheel and leg [3].

Wheels are ideally suited to flat surfaces. However, they are not optimal in soft or rough surfaces. Insect (e.g., roach, spider) locomotion, on the other hand, consists of point contacts and is more suited for these terrains [5]. The Wheg mimics insect mobility in rougher terrain, but uses a design that is consistent with round wheel motion.

To overcome some of the limited mobility shortcomings, a new Wheg was designed for the Surf-zone robot, see Figure 1.2. Specific design will be discussed in Chapter 2.

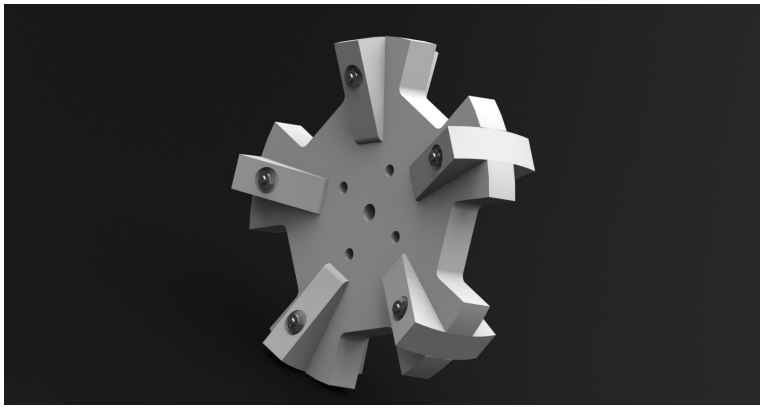


Figure 1.2: LT Bell's Wheg^{TM} design.

1.2 Objectives

The objective was to make a rugged mobile platform for terrestrial operations in the surf-zone. The design of the chassis was an attempt to mitigate problems encountered in previous Surf-zone robot designs. For example, DARc was a rugged platform based on a biolog-

ically inspired modes, which used the Whegs wheel and and a Pelican^{TM} case for payload. The drive controller in DARc was not sufficient for moderate beach climbs. MONTe [3] [4] solved many of the rigidity problems in operation but the drive train failed under high load conditions. AGBOT successfully mimicked the cockroach gate but had poor suspension and physically fell apart after moderate use. This design of DARc, addresses the drive controller problem and adds different wheel designs for testing and measurement. DARc remains modular, like its predecessor.

This thesis addresses the following questions:

- Is this platform capable of operating efficiently in a sea-shore interface environment?
- How does the Wheg design compare to more conventional locomotion methods?
- Is 3-D print material sufficient to withstand use in the marine and rocky environments?

The goal was to determine if the Wheg truly outperformed other wheel types. The DARc has a five leg Wheg, which was compared to a narrow-wheel and a wide-rubber-wheel for performance characteristics.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 2:
Robot Design

2.1 Dynamics and Control

DARc was designed to be a rugged terrestrial model with the ability to exit the surf-zone and navigate various terrains.

Dynamics

The Shuey and Fitzgerald models demonstrated that the dynamical equations of motion for a 3 Degree-of-Freedom (x, y and θ) ground robot are:

$$x(t) = \frac{1}{2}\frac{F_x}{m}t^2 + \dot{x_0}t + x_0 \tag{2.1}$$

$$y(t) = \frac{1}{2}\frac{F_y}{m}t^2 + \dot{y_0}t + y_0 \tag{2.2}$$

$$\theta(t) = \frac{1}{2}\frac{\tau}{I_R}t^2 + \dot{\theta_0}t + \theta_0 \tag{2.3}$$

where $F_x = \frac{\partial U}{\partial x}$, s$F_y = \frac{\partial U}{\partial y}$, U is a function of x and y and I_R is the moment of Inertia. F_x and F_y are treated as constants at each time step.

$$I_R = I_{yy} = \frac{M}{12}(a^2 + c^2)$$

M is the total mass of the robot, a and c are the length and width of the platform, respectively.

Control

In autonomous mode, the ability to control the platform was shown to be a combination of Proportional and Derivative control in a PID controller via the Ziegler-Nichols method:

$$u(t) = .85(1 + \frac{1}{2.5s} + .63s)e(t) \tag{2.4}$$

$u(t)$ is the compensated signal, $e(t)$ is the error as a function at time with respect to a reference signal, and s is a characteristic constant to the platform. The PID coefficients are 0.85, 2.5 and 0.63, respectively.

2.2 Design

Our design evolved from lessons learned on previous models. First, we incorporated the climbing ability of the Wheg with the watertight properties of the Pelican case. Second, our objective was to reduce vibrations and the amount of perforations to the Pelican case. The perforations in the case were used for mounting hardware and caused watertight integrity issues. Additionally, 3-D print material components were sparse printed to be more durable.

The electronics package was changed to overcome difficulties with the motor controllers and to showcase the benefits of a modular construction. The result was an exoskeleton design with modular components, see Figure 2.1.

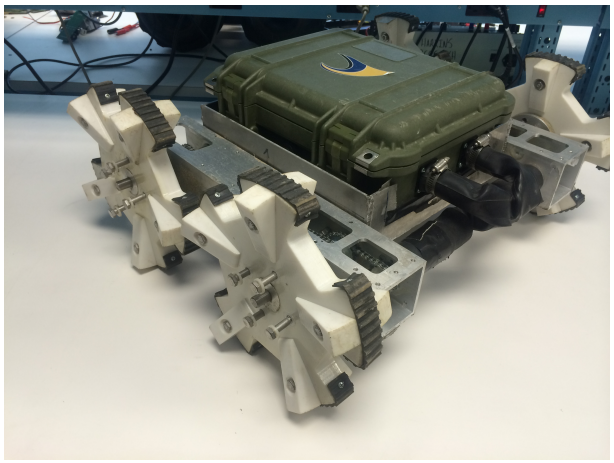


Figure 2.1: DARc's components are modular and can be easily replaced or alternate components can be used.

2.2.1 Base

The Base is a welded 0.318 cm gauge aluminum pan of dimensions 31.9 cm x 30 cm x 7.7 cm (L x W x H), see Figure 2.2. The pan provides the support structure for the chassis and houses the Pelican^{TM} case. This allows the case to have minimal perforations for wiring and facilitates an easier exchange of wheel driving mechanisms. For example, the drive

and suspension system from MONTe could easily be attached to this base with only a slight modification of motor position.

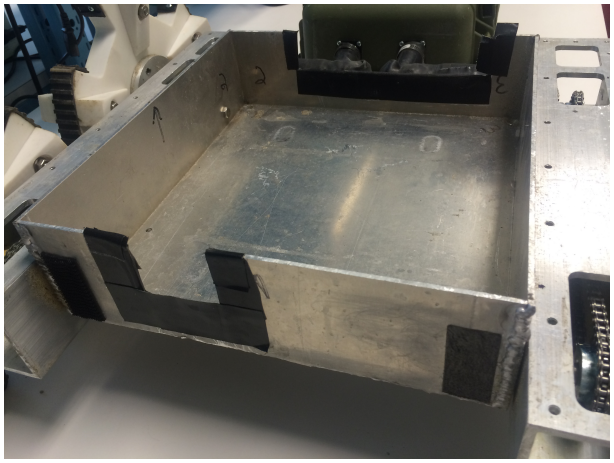


Figure 2.2: The base is designed to receive a Pelican^{TM} 1400 case. It allows the electronics module and the drive assembly to be easily changed in a modular fashion.

2.2.2 Transmission Drives

The transmission drive is a 7.62 cm x 7.62 cm square aluminum tube of 0.318 cm gauge with access ports cut in for maintaining the drive assembly, see Figure 2.3. The motor drives one stainless steel chain to the lead wheel shaft and another chain is connected from the lead wheel shaft to the rear wheel shaft. This results in tank-style steering control. There are two sealed bearings that support each shaft. The tubing encloses the chain mechanism preventing foreign debris from becoming entangled in the chain. The intent was to use a clear acrylic sheet to cover the access ports using silicon as a gasket material for waterproofing. End plugs will be designed, in 3-D print or nylon material, to prevent water from entering the transmission.

2.2.3 Chassis

The transmission mounts to the base, completing the chassis, see Figure 2.4. The 10.16 cm hubs were designed to fit a commercial rubber tire. All 3-D print wheels were designed to fit this pattern allowing for interchangeability. The wheel base of the robot measures 28.8 cm from the side and 56.3 cm from the front.

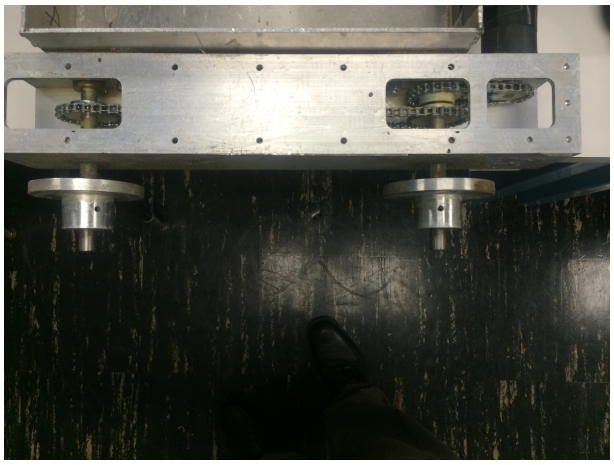


Figure 2.3: The square aluminum tube houses the drive assembly. The motor is mounted to the outside of the tube. Each transmission is independent of the other.

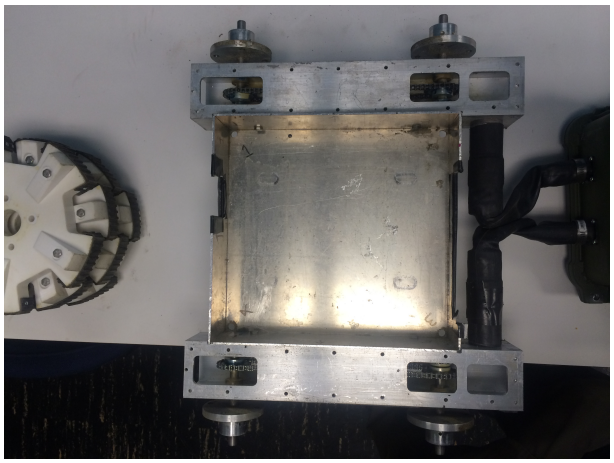


Figure 2.4: The chassis is formed of three separate components, the two transmissions and the base, allowing for easy system modification.

2.2.4 Motors

We chose to use IG-42 24 VDC 122 RPM Brush Gears Motor 2.5. They were mounted in the forward portion of the robot to assist with center balance in climbing situations, see Figure 2.6.

2.2.5 Waterproof Housing

The Pelican 1400 case has exterior dimensions of 33.96 cm x 29.52 cm x 15.24 cm (L x W x H) and interior dimensions of 29 cm x 22.53 cm x 13.16 cm (L x W x H); it is crush

Figure 2.5: Brushed permanent magnet DC motor, variable speed and reversible, 24VDC, Reduction Ratio: 1:49, Rated Torque: 16 kgf-cm, Rated Speed: 122 RPM, and no Load Current: $<$ 500mA, from [7].

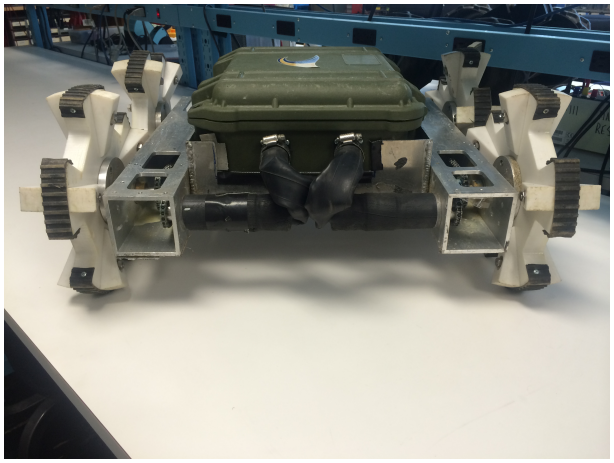


Figure 2.6: The motors were installed in the forward portion of the robot and were placed in a protected area. The rubber sleeve acts as a waterproof covering.

proof and watertight to approximately one meter, see Figures 2.7 and 2.8. This case has been used with success in the last three iterations of the Surf-zone robot and has been an excellent product.

2.2.6 Controller

DARc uses a Roboteq^{TM} HDC2450 motor controller that provides up to 150 amps of peak current capacity, see Figure 2.9. It is programmable in BASIC and is capable of interfacing with sensor and navigation packages that assist autonomy, including the RIO^{TM}/Raspberry

Figure 2.7: The Pelican 1400 case houses the electronic module.

Figure 2.8: The interior has ample room for the controller, batteries and RC module.

Pi^{TM} system as a micro-controller, see Figure 2.10.

The transition from RC to autonomous control is quite straightforward with the Roboteq controller. RC control was used to gather qualitative data over various terrains. Autonomous mode was employed for data analyses in a controlled environment.

2.2.7 Wheels

We chose three wheel types to test. A new Wheg designed by LT Bell, a round wheel that was similar to the Wheg in material and size, and a commercial rubber wheel that was much wider than the other two.

Figure 2.9: The Roboteq^{TM} HDC2450 is a Brushed DC Motor Controller, Dual Channel, 150A, 50V, Encoder in, USB, CAN capable with on-board 32-bit microcomputer and uses Micro Basic as the language, from [8].

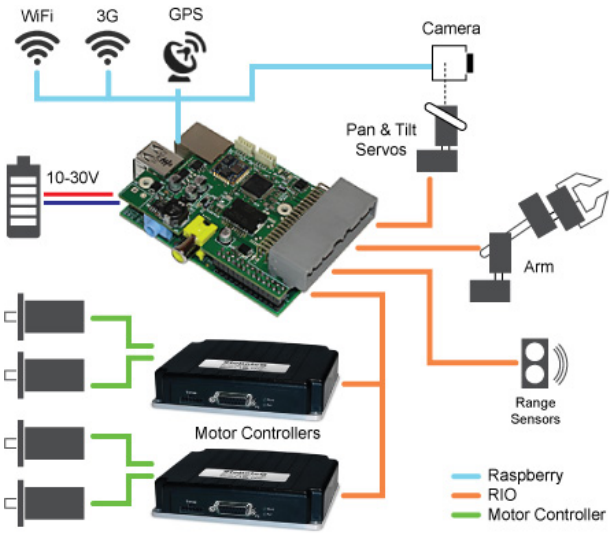


Figure 2.10: This figure depicts some of the additional capability of the Roboteq controller, from [9].

Wheg

The DARc Wheg was designed out of sparse print 3-D material as a fast prototype technique and is shown in Figure 2.11. It has five legs that minimize rotation vibration when compared to other models with fewer legs, while leaving enough gap for purchase on uneven surfaces. The cross member provides for weight distribution on beach surfaces and acts as a paddle in a softer terrain such as mud or sand. The sparse print technique was used to lengthen mean-time-between failure parameters for the wheel. Solid print poly-carbon printed wheels failed often due to grain structure inherent in the print process.

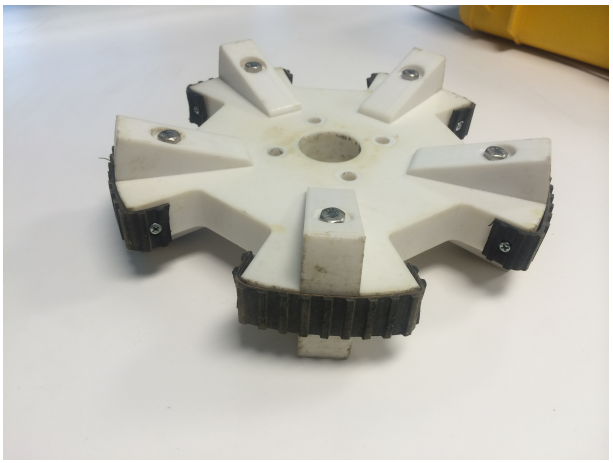


Figure 2.11: 25.4 cm diameter by 2.54 cm thick five leg Wheg with automotive timing belt as tread designed by LT Bell.

This edition of the Wheg was modeled in SolidWorks^{TM} Computer Aided Design (CAD) software. The outer diameter is 12.7 cm by 2.54 cm thick and has five legs with automotive timing belt as tread. The inner diameter is 8.5 cm which gave the Wheg 4.2 cm space for gripping obstacles. The paddle structure is designed to be replaceable in the event of breakage. It was also designed with a sufficient angle to prevent wet sand or mud from collecting on the wheel, see Figure 2.12.

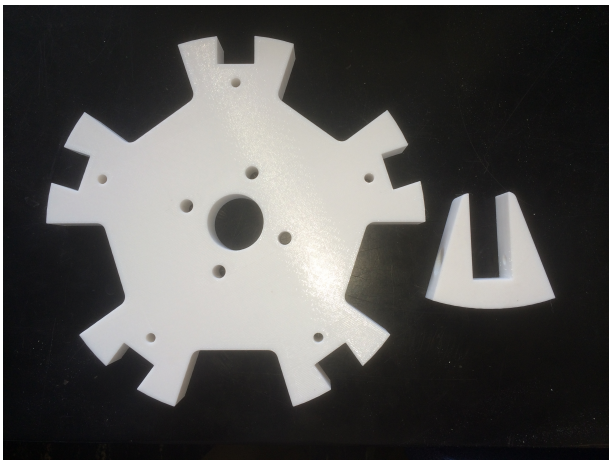


Figure 2.12: This Wheg was designed with paddle elements to move through sandy and muddy environments while avoiding collection of debris.

Initial testing showed this Wheg has excellent climbing capability. It will climb a standard stair with ease. However, DARc will flip backward often since it cannot deploy a tail yet.

The goal is to climb an obstacle the same height as the diameter of the wheel.

Round Wheel

The round 3-D print wheel (Figure 2.13) was designed, and sparse printed, to the same diameter and thickness specification as the Wheg. The round wheel was used for baseline measurements and for field experiments that demonstrated mobility. Current and power comparisons to the Wheg and the rubber wheel were also conducted.

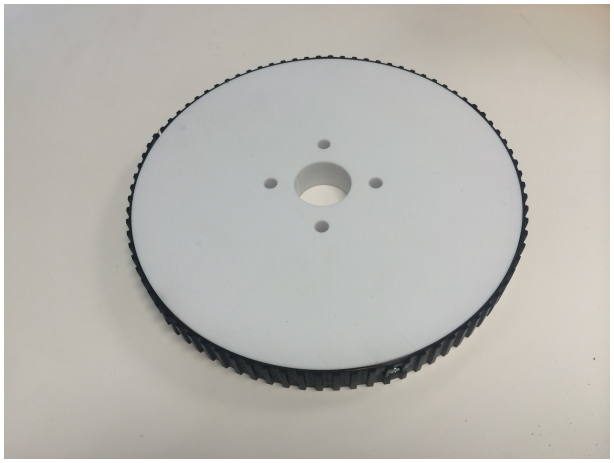


Figure 2.13: 25.4 cm diameter by 2.54 cm thick round wheel with automotive timing belt as tread.

Rubber Wheel

DARc also employed a commercial rubber wheel for performance tests and measurements in comparison with the other wheels, see Figure 2.14. The tire is a nylon tube type 4.10/3.50-4 tire that has been used in other local designs.

2.2.8 Tail

The concept of a lobster-like tail was employed by MONTe for climbing assist and stability in rugged environment, see Figure 2.15. The end goal is to employ a similar tail structure on DARc for future mobility tests.

Figure 2.14: A commercial rubber nylon tube type 4.10/3.50-4 tire to test how a wider tire operates in the surf-zone environment.

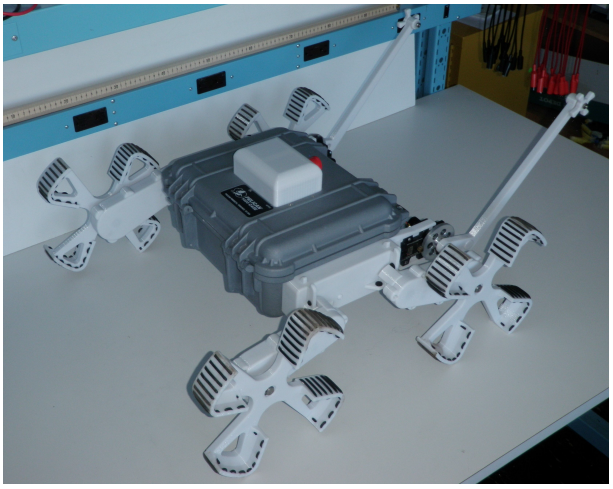


Figure 2.15: Depicted is the previous Surf-zone robot, MONTe, which illustrates an excellent example of a retractable tail for righting and climbing situations, from [3].

CHAPTER 3:
Experiments and Results

3.1 Data Collection

Our objective was to test DARc under different load conditions to compare power performance against mobility. The approach was to collect qualitative and quantitative data under RC and Autonomous control, respectively.

3.1.1 Fixed-Route

Quantitative data was collected for the different wheel designs by measuring performance on a fixed-route plan as shown in Figure 3.1 This pattern was designed to test a full-complement of motion scenarios with all the wheel types (Wheg, round, and rubber) on all surface types (grass, concrete, and beach). These runs included:

- a 90-degree zero speed pivot turn to the left[1] followed by a 90-degree pivot turn right with forward momentum.
- A 90-degree Ackermann[2] turn right where the left wheels are full powered and the right are half powered
- a 180-degree tank turn left where the right wheels are full power forward and the left wheels are full power back.

This pattern is repeated but with opposite wheel direction bias to get the full range of testing. Time loops were coded into the Roboteq controller using Micro-Basic and can be found in Appendix A.

3.1.2 Modem

DARc was placed at the start position and with different wheel configurations and tested with 5 runs each for each wheel design. Time stamped Current data was collected and sent to the remote station as raw data for post run analysis, refer to Figure 3.2. This was

[1]The pivot turn to the left is accomplished by keeping the right motor unpowered while the left wheels are full speed.

[2]This is like a car turn.

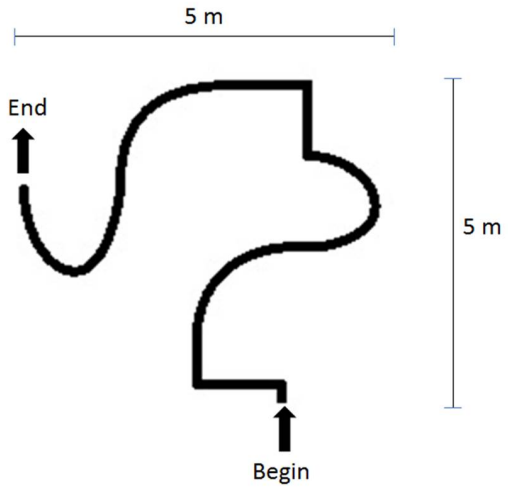


Figure 3.1: Experimental Route Plan.

accomplished with an RS232 serial communication protocol via a Proxim^{TM} Modem as shown in Figure 3.3

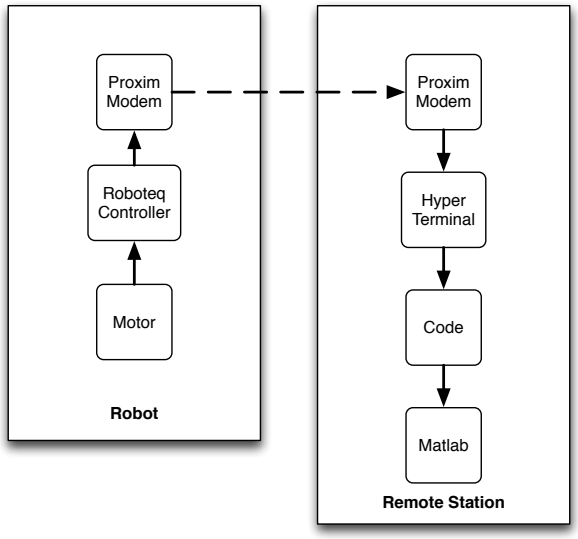


Figure 3.2: Current data from the Roboteq is sent wirelessly to a laptop and processed by MATLAB to get desired graphs.

3.1.3 Controller

On the robot side, the Roboteq controller (Figure 2.9) has a monitoring system, that when invoked, can monitor motor currents up to 150 Amps. A Micro-Basic script was written to

Figure 3.3: The Proxim^{TM} modems are capable of 11540 baud and can use serial or Ethernet connections.

read left and right motor currents during an experimental run. This data was sent through the Proxim modem at 11540 baud to a Windows HyperTerminal session. Because our data stream was session-less, we encountered dropped packets. For our 100 Hz sampling rate, dropped packets were not a problem, but they tended to *corrupt* the symmetry of our data-sets. These problems were remedied by a script written in C-Sharp by Dr. Keith Cohn, NPS, which can be found in Appendix A.

3.2 Data Reduction and Calculations

The raw data sets had high frequency noise, Figure 3.4, (jitter) in the time domain. To manage this, the data was transformed to the frequency domain via Fast Fourier Transform, filtered and then transformed back to the time domain. The objective was to smooth out the jitter prior to calculations.

Discrete Fast Fourier Transfiorm

The FFT transforms a signal from one domain to another. For example, to transform a function $f(t)$ from the time domain to the frequency domain $F(f)$ we have:

$$F(f) = \int_{-\infty}^{\infty} f(t) e^{2\pi i f t} dt \tag{3.1}$$

To transform back to the time domain, from the frequency domain, you reverse this process:

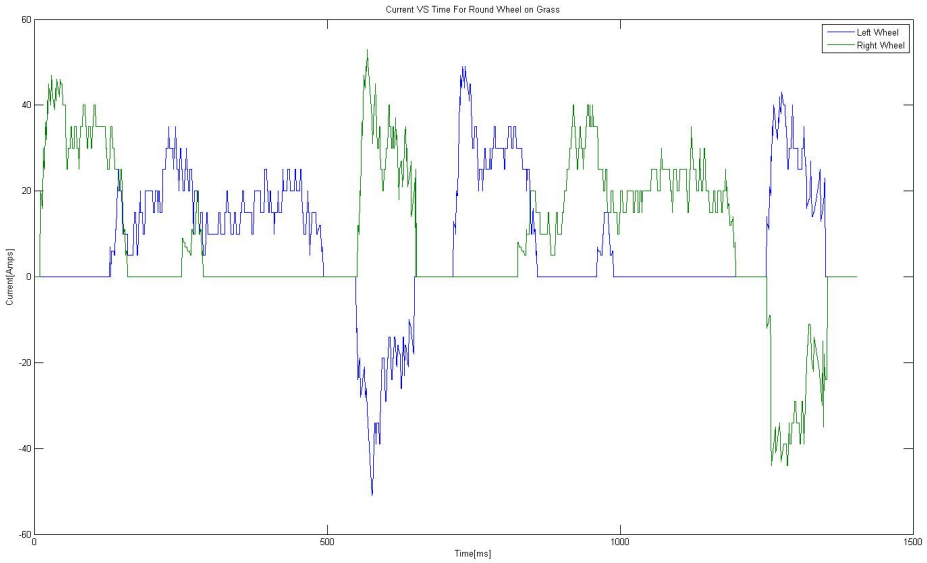


Figure 3.4: Raw Current Data For Round Wheel in Grass.

$$f(t) = \int_{-\infty}^{\infty} F(f) e^{-2\pi i f t} df \tag{3.2}$$

Since our data sets are discrete, the discreet forms are applicable:

$$X(n) = \frac{1}{N} \sum_{k=0}^{N-1} x(k) e^{2\pi i k n / N} \tag{3.3}$$

and

$$x(n) = \sum_{k=0}^{N-1} X(k) e^{2\pi i k n / N}. \tag{3.4}$$

Here, our samples $x(n)$ are N periodic and we are only interested in the *real* part.

Since our primary signal was on the order of Hz, in the frequency domain, we used a Rectangular Window to filter high-frequency components.

Sample Data

To illustrate, raw current data is shown in Figure 3.5, and after signal processing the data with the FFT/IFFT filter, we observe the result in Figure3.6.

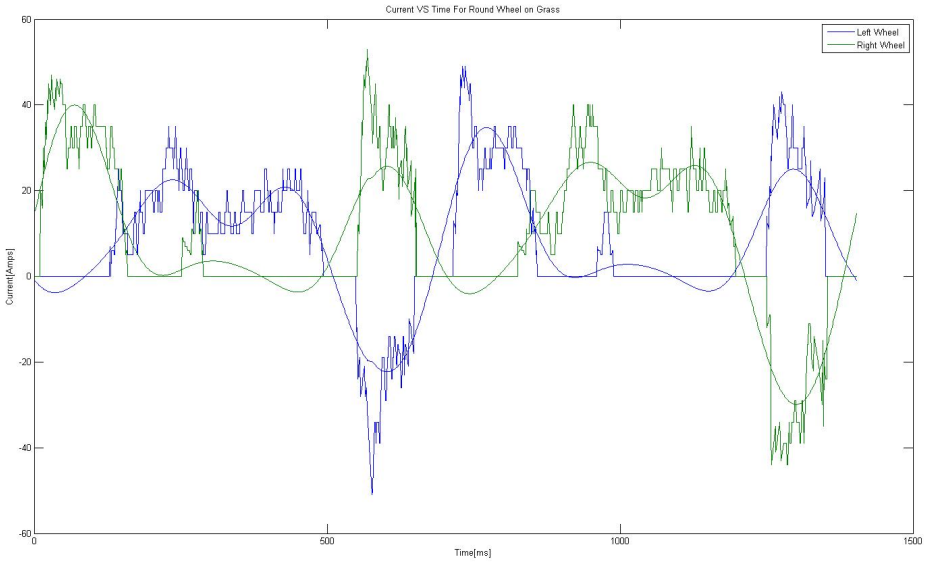


Figure 3.5: Raw Current Data with Filtered Curve Overlayed

The resultant motor current use, for the various runs, was then calculated and reported via standard methods. The sum N of j instances were determined: the mean $< j >$ calculated and finally the mean squared $< j^2 >$ was determined. From these, we get the standard deviation σ as shown.

The Mean and Standard Deviation

$$N = \sum_{j=0}^{\infty} N(j) \tag{3.5}$$

$$< j >= \frac{\sum jN(j)}{N} \tag{3.6}$$

$$< j^2 >= \frac{\sum_{J=0}^{\infty} j^2 N(j)}{N} \tag{3.7}$$

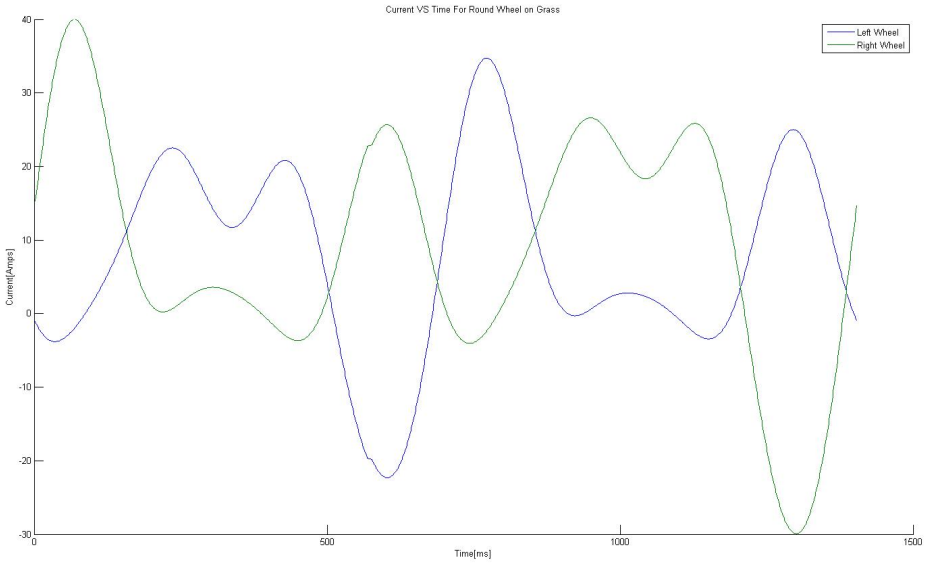


Figure 3.6: The data has been processed and now has a smooth curve.

Finally, $\sigma = \sqrt{< j^2 > - < j >^2}$ is calculated as usual. In these expressions, j is an instance of the measured current.

The power bus for the Robot was set at 24 volts. The average power P_{ave} was then simply:

$$P_{\text{ave}} = V \cdot I_{\text{ave.}}$$

3.3 Experimental Tests and Results

Mobility of the robot was tested against a Figure Pattern, Figure 3.1, in different terrains. Current metrics were collected, and processed as discussed above. The results for each are summarized below. Refer to Appendices B and C for the filtered and raw data sets.

3.3.1 Grass

The grass terrain provided the most consistent results. Little wheel slippage was observed. Table 3.3.1 shows that the peak current for the round wheel was approximately 40 amps. The time average current draw was 12.07 Amps with a standard deviation $\sigma = \pm$ 0.18

Wheel	Peak Current (Amps)	Average Current (Amps)	Average Power (Watts)
Round Wheel	45	12.07 ± 0.18	290
Rubber Wheel	45	11.85 ± 0.30	284
Wheg	60	13.11 ± 0.25	315

Table 3.1: Peak Current, Average Current and Power for Wheels on Grass.

Amps over five runs, see Figure B.1. The rubber wheel drew current similar to the round wheel. Its peak current was approximately 40 amps. The time average current draw was 11.85 Amps with a standard deviation $\sigma = \pm$ 0.30 Amps, see Figure B.2. The Wheg peak was 45 Amps. The time average current draw was 13.11 Amps with a standard deviation $\sigma = \pm$ 0.25 Amps, see Figure B.3.

3.3.2 Concrete

Wheel	Peak Current (Amps)	Average Current (Amps)	Average Power (Watts)
Round Wheel	40	9 ± 0.22	198
Rubber Wheel	50	13.65 ± 0.17	328
Wheg	40	9.8 ± 0.27	235

Table 3.2: Peak Current, Average Current and Power for Wheels on Concrete.

The concrete surface testing was designed to mimic an urban or hard surface terrain. We expected the rubber wheel to perform the best in this environment. This surface is particularly hard on the Wheg because its non-circular shape caused heavy vibrations to be transmitted throughout the robot.

The peak current for the round wheel was approximately 28 amps. The time average current draw was 9.00 Amps with a standard deviation $\sigma = \pm$ 0.22 Amps, see Figure B.4. These numbers are low. There was a considerable amount of slippage observed during the tests.

The peak current for rubber wheel was approximately 40 amps. The time average current draw was 13.65 Amps with a standard deviation $\sigma = \pm$ 0.17 Amps, see Figure B.5. The rubber wheel performed the best on this surface.

The Wheg peak was 33 Amps. The time average current draw was 9.80 Amps with a standard deviation $\sigma = \pm$ 0.27 Amps, see Figure C.7. These numbers are close to those

of the round wheel. Their materials were the same 3-D print and automotive timing belt. The peak was much higher due to the torque involved in the lifting action of the Wheg's rotation. Consider it as a rolling pentagon.

3.3.3 Sand

Wheel	Peak Current (Amps)	Average Current (Amps)	Average Power (Watts)
Round Wheel	45	19.56 ± 0.41	469
Rubber Wheel	45	20.65 ± 2.15	496
Wheg	60	26.79 ± 0.39	643

Table 3.3: Peak Current, Average Current and Power for Wheels on Sand.

Sand testing proved the most difficult environment. For all wheels, the sand required more average power to maneuver.

We assumed the round wheel would perform poorly because it is narrow and that the rubber wheel would do better because it is wide and has more displacement. Infact, both wheels performed poorly and drew more current than expected. However, the rubber wheel did stay on top of the soft sand

The peak current for the round wheel was approximately 45 amps. The time average current draw was 19.56 Amps with a standard deviation $\sigma = \pm$ 0.41 Amps, see Figure B.7. This wheel had a tendency to dig down until it hit more compact sand. It traveled better than the rubber wheel, but failed in comparison to its grass test performance.

The peak current for the rubber wheel was approximately 50 amps. The time average current draw was 20.65 Amps with a standard deviation $\sigma = \pm$ 2.15 Amps, see Figure B.8. This wheel slipped a tremendous amount and failed to conduct the pattern significantly. Having encoder data here would have been a tremendous help. Qualitatively, it was the worst wheel in this environment. It is only suited for hard and flat surfaces.

The Whegpeak was 62 Amps. The time average current draw was 26.79 Amps with a standard deviation of $\sigma = \pm$ 0.39 Amps, see Figure B.9. The paddle shape in the Wheg allowed it to have more purchase in the loose material and it traveled farther in comparison to the round and rubber wheel.

3.3.4 Hill

Wheel	Peak Current (Amps)	Average Current (Amps)	Average Power (Watts)
Round Wheel	45	21.74 ± 0.34	522
Rubber Wheel	45	26.94 ± 0.30	647
Wheg	60	23.21 ± 0.88	557

Table 3.4: Peak Current, Average Current and Power for Wheels on Hill.

We chose a uniform grassy hill to see what kind of current was drawn in steep conditions. As in the the grass test, the round and rubber wheels were similar in comparison.

The peak current for the round wheel was approximately 45 amps. The time average current draw was 21.74 Amps with a standard deviation $\sigma = \pm$ 0.34 Amps over five runs, see Figure B.10.

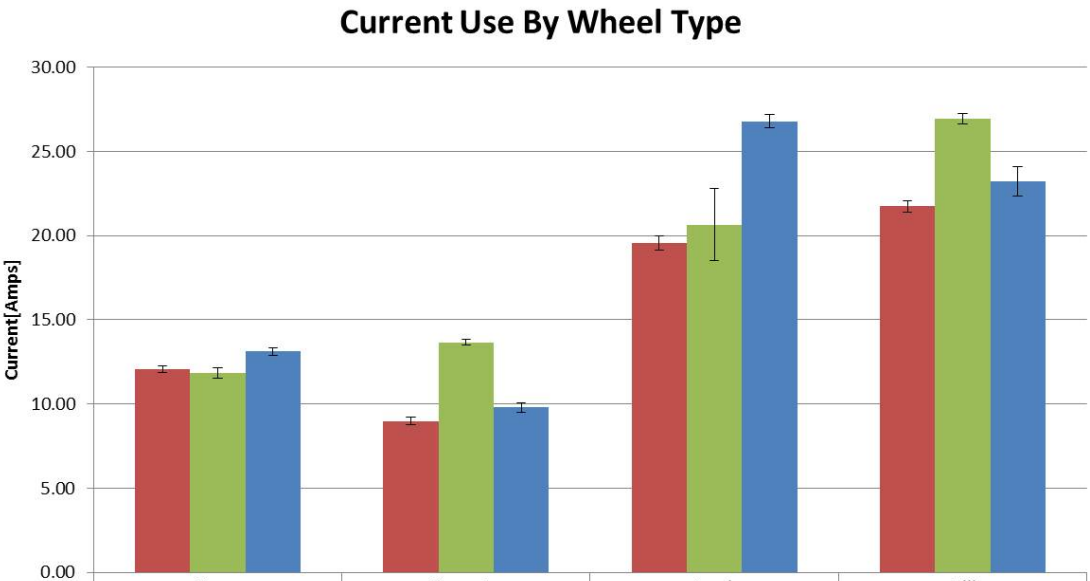
The rubber wheel drew slightly more current as it had a wider contact area with the ground. Its peak current was approximately 45 amps. The time average current draw was 26.94 Amps with a standard deviation $\sigma = \pm$ 0.30 Amps,see Figure B.11.

The Wheg peak was 44 Amps. The time average current draw was 23.21 Amps with a standard deviation $\sigma = \pm$ 0.88 Amps, see Figure B.12.

3.4 Current Use Summary

All wheel types seemed to use a similar amount of current on a given surface. This shows that choosing a more mobile wheel, such as the Wheg, will not impact the on-station time of the robot, see Figure 3.7.

It was interesting to note that the rubber wheel performed best on concrete because it had no slippage and therefore drew more current. The Wheg seemed to perform better in the soft sand and it drew more current than the others. Encoder data and navigation plots are needed to confirm the idea that less slippage would use more current and provide better forward motion.



	Grass	Concrete	Sand	Hill
Round	12.07	9.00	19.56	21.74
Rubber	11.85	13.65	20.65	26.94
Wheg	13.11	9.80	26.79	23.21

Figure 3.7: Current Use Summary by Surface Type.

3.4.1 The Wheg

The Wheg was a central design component of this platform. It has proven to be successful. In comparison to the other wheels, it performed the same as the round and rubber wheel in general locomotion. It surpasses the two other wheels in climbing capability and does moderately better in the loose sand environment. The major drawback for the Wheg is the excessive vibration and stress on the chain. Additionally, the Wheg design can successfully navigate vegetation and four-inch stairs.

3.4.2 The Round Wheel

The round wheel was specifically designed as a comparison to the Wheg in dimensions and material. It performed as expected in most situations. It failed to climb a four-inch step, but did surprisingly well on the ice plant.

3.4.3 The The Rubber Wheel

The rubber wheel was used as a commercially available possibility. We expected that it was wide enough to stay on top of the loose sand. It did not perform well enough to be considered as suitable for the Surf-zone robot.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 4:
Conclusion

This edition of the the Surf-Zone robot combined successful ideas from previous models and improved upon them. This included the addition of modularity, an improved drive train and the ability to quickly change out wheel models based on expected terrain environments. Follow-on work will necessarily require the addition of a tail to assist stability and climb capabilities.

4.1 Platform Capability

The design has the appropriate size and weight for use by Special Forces personnel. The build cost is low enough to allow for disposability. Modularity allows for multiple mission capability by simply changing the electronics package or the drive train assembly. Future models will need to include a suspension to reduce the wear on drive mechanisms if the Wheg^{TM} is the desired wheel.

4.1.1 Wheel Assessment

The data indicate that required power for mobility in the different terrains was, mostly, wheel independent. The conclusion, then, is that efforts should be focused on the best wheel for overall mobility. It is also concluded, based upon the beach hill climb data, the the Wheg performed better across the scope of terrain.

The Wheg^{TM} preformed comparably to the round wheels in hard or grassy area. It outperformed in moderate sand, dense vegetation and small rocky areas. The LT Bell edition of the Wheg^{TM} was able to climb four-inch stairs with ease. This could be considered for search and rescue in areas such as smoky buildings or contaminated areas such as the Fukushima Daiichi Nuclear Power Plant disaster. For this prototype, waterproofed sparse print 3-D print material was acceptable.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A: Basic Autonomous Pattern Run

```
\begin{lstlisting}[breaklines]
''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''
'Program Title: Autonomous Pattern Run
'Programmer:  Timothy L. Bell
'Coding Language: Micro-Basic
'
'Supervisor:  Prof Harkins
'
'Program Description:   This program will run a set pattern and send
'data parameters from the motor controller.
''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''

print("starting ... \r")
print("Time\t","RAmp\t","LAmp\t","RPwr\t","LPwr\t", "RRpm\t",
"LRpm\t","REncoder\t", "LEncoder\t", "\n") 'Prints column headers

dim i as integer
i = 0 'Used as time counter.  Each is 10 ms.
if i > 1700 then i = 0

top:

LAmp = getvalue(_MOTAMPS, 2)  'Gets left motor amperage value
RAmp = getvalue(_MOTAMPS, 1)  'Gets right motor amperage value

LPower = getvalue(_MOTPWR, 2)  'Gets left motor power value
RPower = getvalue(_MOTPWR, 1)  'Gets right motor power value

LRpm = getvalue(_ABSPEED, 2)  'Gets left motor speed value
RRpm = getvalue(_ABSPEED, 1)  'Gets right motor speed value
```

```
LEncoder = getvalue(_ABCNTR, 2) 'Gets left motor encoder value
REncoder = getvalue(_ABCNTR, 1) 'Gets left motor encoder value

print(i,"\t",LAmp,"\t",RAmp,"\t",LPower,"\t",RPower,"\t",LRpm,"\t",
RRpm,"\t",LEnc,"\t", REnc,"\n\r") 'Prints back data

i = i + 1 'Starts I count loop increase by 1 in 10 ms steps.

if i < 100 then '90 degree Left Pivot Turn
SetCommand(_GO, 1, 1000)'1 is right motor.
Setcommand(_GO, 2, 0)  '2 is left motor
end if

if i >= 100 and i < 150 then
SetCommand(_GO, 1, 1000)
SetCommand(_GO, 2, 1000)
end if

if i >= 150  and i < 230 then '90 degree Right Pivot Turn
SetCommand(_GO, 1, 0)
setcommand(_GO, 2, 1000)
end if

if i >= 230  and i < 280 then 'Straight Run
SetCommand(_GO, 1, 1000)
setcommand(_GO, 2, 1000)
end if

if i >= 280  and i < 480 then '90 degree Right Ackermann Turn
SetCommand(_GO, 1, 500)
setcommand(_GO, 2, 1000)
end if
```

```
if i >= 480  and i < 535 then 'All Stop
SetCommand(_GO, 1, 0)
setcommand(_GO, 2, 0)
end if

if i >= 535 and i < 615 then '180 degree Tank Turn Left
SetCommand(_GO, 1, 1000)
setcommand(_GO, 2, -1000)
end if

' Robot begins second leg of test run conducting
a pattern opposite to the first leg

if i >= 615 and i < 700 then 'All Stop
SetCommand(_GO, 1, 0)
Setcommand(_GO, 2, 0)
end if

if i >= 700 and i < 800 then '90 degree Pivot Turn Right
SetCommand(_GO, 1, 0)
Setcommand(_GO, 2, 1000)
end if

if i >= 800 and i < 850 then 'Straight run
SetCommand(_GO, 1, 1000)
SetCommand(_GO, 2, 1000)
end if

if i >= 850  and i < 930 then '90 degree Pivot Turn Left
SetCommand(_GO, 1, 1000)
setcommand(_GO, 2, 0)
end if

if i >= 930  and i < 980 then 'Straight Run
```

```
SetCommand(_GO, 1, 1000)
setcommand(_GO, 2, 1000)
end if

if i >= 980  and i < 1180 then '90 degree Ackermann Turn Left
SetCommand(_GO, 1, 1000)
setcommand(_GO, 2, 500)
end if

if i >= 1180  and i < 1235 then'All Stop
SetCommand(_GO, 1, 0)
setcommand(_GO, 2, 0)
end if

if i >= 1235 and i < 1315 then '180 Tank Turn Right
SetCommand(_GO, 1, -1000)
setcommand(_GO, 2, 1000)
end if

if i >= 1315 and i <1400 then 'All Stop
SetCommand(_GO, 1, 0)
Setcommand(_GO, 2, 0)
end if

if i >= 1405 then terminate 'Ends Pattern Run

wait(10) 'Pause 10  ms
goto top 'Return to top

''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''
'Program Title: Autonomous Hill Run
'Programmer:  Timothy L. Bell
'Coding Language: Micro-Basic
'
```

```
'Supervisor:  Prof Harkins
'
'Program Description:
This program will run for a preset period of time and send
'data parameters from the motor controller.
''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''

print("starting ... \r")
print("Time\t","RAmp\t","LAmp\t","RPwr\t","LPwr\t", "RRpm\t",
"LRpm\t","REncoder\t", "LEncoder\t", "\n") 'Prints column headers

dim i as integer
i = 0 'Used as time counter.  Each is 10 ms.
if i > 1700 then i = 0
wait (10) 'Pause 10 ms

top:

LAmp = getvalue(_MOTAMPS, 2)  'Gets left motor amperage value
RAmp = getvalue(_MOTAMPS, 1)  'Gets right motor amperage value

LPower = getvalue(_MOTPWR, 2)  'Gets left motor power value
RPower = getvalue(_MOTPWR, 1)  'Gets right motor power value

LRpm = getvalue(_ABSPEED, 2)  'Gets left motor speed value
RRpm = getvalue(_ABSPEED, 1)  'Gets right motor speed value

LEncoder = getvalue(_ABCNTR, 2) 'Gets left motor encoder value
REncoder = getvalue(_ABCNTR, 1) 'Gets left motor encoder value

print(i,"\t",LAmp,"\t",RAmp,"\t",LPower,"\t",RPower,"\t",
LRpm,"\t",RRpm,"\t",LEnc,"\t", REnc,"\n\r") 'Prints back data

i = i + 1 'Starts I count loop and increases by 1 in 10 ms steps
```

```
if i >= 50 and i < 550 then 'Straight five second run
SetCommand(_GO, 1, 1000)
SetCommand(_GO, 2, 1000)
end if

if i >= 555 then terminate

wait(10) 'Pause 10  ms
goto top 'Return to top

''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''
'Program Title: RC Data Collection
'Programmer:  Timothy L. Bell
'Coding Language: Micro-Basic
'
'Supervisor:  Prof Harkins
'
'Program Description:
This program will data parameters from the motor
'controller while operating in Remote Control(RC)
'mode.
''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''

print("starting ... \r")
print("Time\t","RAmp\t","LAmp\t","RPwr\t","LPwr\t", "RRpm\t",
"LRpm\t","REncoder\t", "LEncoder\t", "\n") 'Prints column headers

dim i as integer
i = 0 'Used as time counter.  Each is 10 ms.

top:

i = i + 1 'Starts I count loop and increases by 1 in 10 ms steps
```

```
LAmp = getvalue(_MOTAMPS, 2)  'Gets left motor amperage value
RAmp = getvalue(_MOTAMPS, 1)  'Gets right motor amperage value

LPower = getvalue(_MOTPWR, 2)  'Gets left motor power value
RPower = getvalue(_MOTPWR, 1)  'Gets right motor power value

LRpm = getvalue(_ABSPEED, 2)  'Gets left motor speed value
RRpm = getvalue(_ABSPEED, 1)  'Gets right motor speed value

LEncoder = getvalue(_ABCNTR, 2) 'Gets left motor encoder value
REncoder = getvalue(_ABCNTR, 1) 'Gets left motor encoder value

print(i,"\t",LAmp,"\t",RAmp,"\t",LPower,"\t",RPower,"\t",LRpm,"\t",
RRpm,"\t",LEnc,"\t", REnc,"\n\r") 'Prints back data

wait(10) 'Pause 10  ms
goto top 'Return to top

''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''
'Program Title: Text File Converter
'Programmer:  Dr. Keith Cohn
'Coding Language: C Sharp
'
'Program Description:
This program cleans collected data by deleting
data lines that contain errors such as nulls.
''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''

using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Text;
```

```
namespace TextFileConverter
    {
    class Program
        {
        static void Main(string[] args)
            {
            if (args.Length < 1)
                {
                Console.WriteLine("You must specify a filename.");
                return;
                }

            List<string> lines = readFile(args[0]);

            FileInfo fi = new FileInfo(args[0]);
            string filename =
            System.IO.Path.GetFileNameWithoutExtension(fi.Name);
            filename += "_converted.txt";

            saveFile(filename, lines);
            }

        // read file
        static private List<string> readFile(string filename)
            {
            FileStream fs = null;
            TextReader tr = null;
            List<string> lines = new List<string>();
            char[] delimit = { ' ', '\t' };


            try
                {
```

```
        fs = new FileStream(filename,
        FileMode.Open, FileAccess.Read);
        tr = new StreamReader(fs);

        string[] columns;
        string line;

        tr.ReadLine();

        line = tr.ReadLine();
        columns = line.Split(delimit,
        StringSplitOptions.RemoveEmptyEntries);
        int columnCount = columns.Length;

        while ((line = tr.ReadLine()) != null)
            {
            columns = line.Split(delimit,
            StringSplitOptions.RemoveEmptyEntries);
            if (columns.Length == columnCount)
                lines.Add(line);
            }
        }

    catch { Console.WriteLine
    ("The was an error processing the file."); }
    finally
        {
        if (fs != null)
            tr.Close();
        }

    return lines;
    }
// save file
```

```
        static private void saveFile(string filename, List<string> lines)
            {
            FileStream fs = null;
            TextWriter tw = null;

            try
                {
                fs = new FileStream(filename, FileMode.Create,
                FileAccess.Write);
                tw = new StreamWriter(fs);

                foreach (string line in lines)
                    tw.WriteLine(line);
                }
            catch { Console.WriteLine
            ("The was an error processing the file."); }
            finally
                {
                if (fs != null)
                    tw.Close();
                }
            }

        }
    }
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 Program Title: Noisy-to-Smooth data curve
 Programmer:  Prof Harkins and Timothy L. Bell
 Coding Language: MATLAB

 Supervisor:  Prof Harkins

 Program Description:  This program transform raw data to a smooth
```

```
  curve.
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all
x = load ('mayroundgrassrun1.txt'); %loads desired file

t = x(:,1);
a = x(:,2);
b = x(:,3);
f = fft(a);
g = fft(b);

NN = 8; %adjust to fit desired curve to raw data
f(NN:(length(a)-NN)) = 0;
g(NN:(length(b)-NN)) = 0;
e = real(ifft(f));
h = real(ifft(g));
mean_right = mean(a);
mean_left = mean(b);
avgcurrent = (mean_right + mean_left)/2; %average current draw for run
fprintf ('Average Current is: %.2f Amps\n ', avgcurrent);
plot(t,a, t,b); %plots raw data
hold on
plot(t,e, t,h); %plots FFT smooth data
xlabel('Time[ms]');
ylabel('Current[Amps]');
title('Current VS Time For Round Wheel on Grass');
legend('Left Wheel','Right Wheel');
hold off
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B:
Filtered Current Data

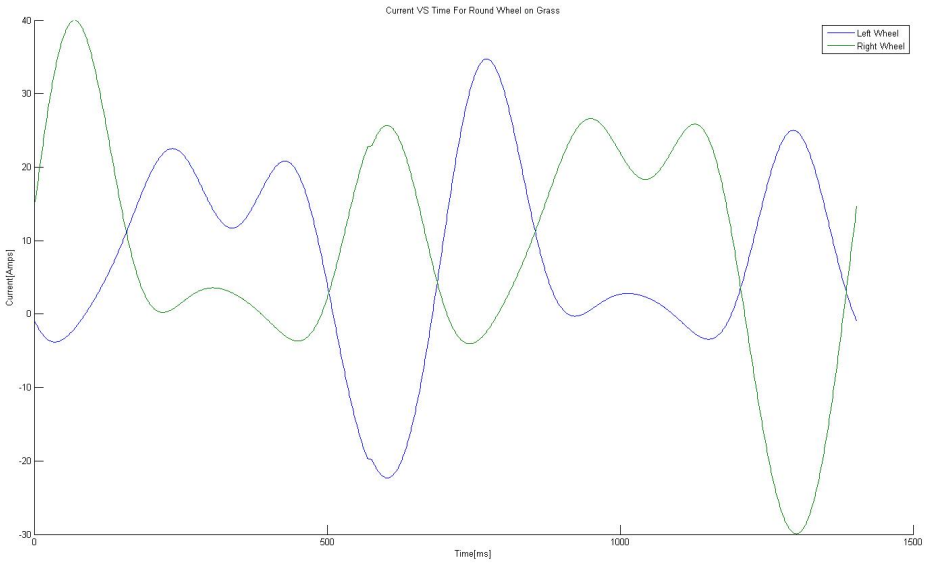


Figure B.1: Round Wheel Current on Grass.

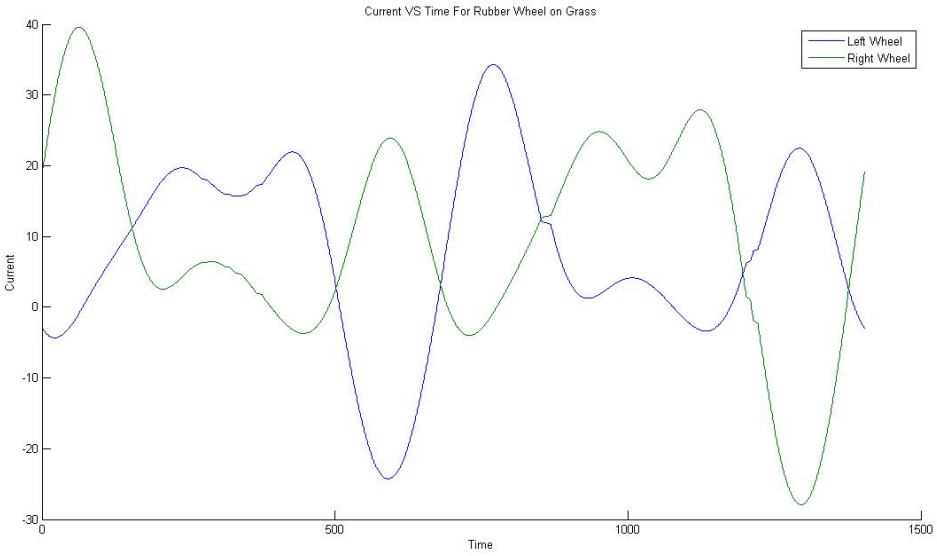


Figure B.2: Rubber Wheel Current on Grass.

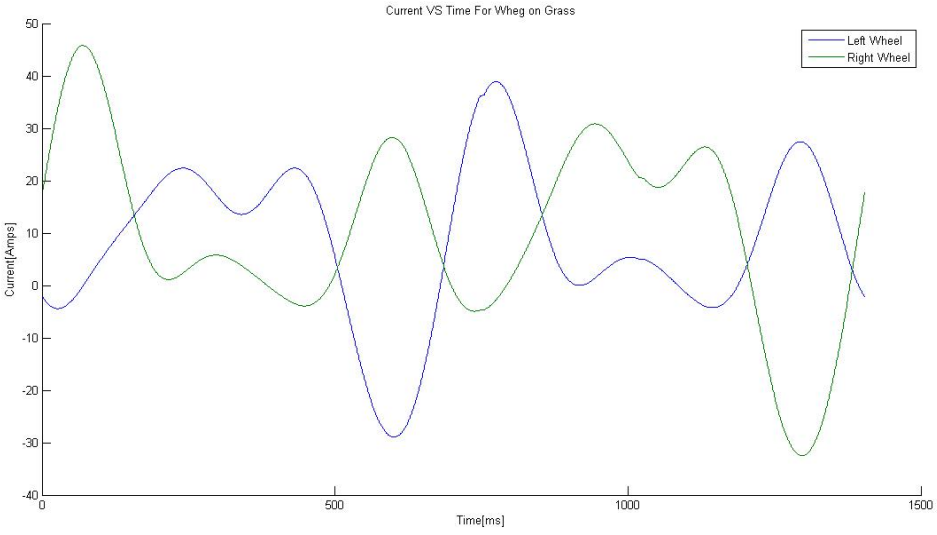


Figure B.3: Wheg Current on Grass.

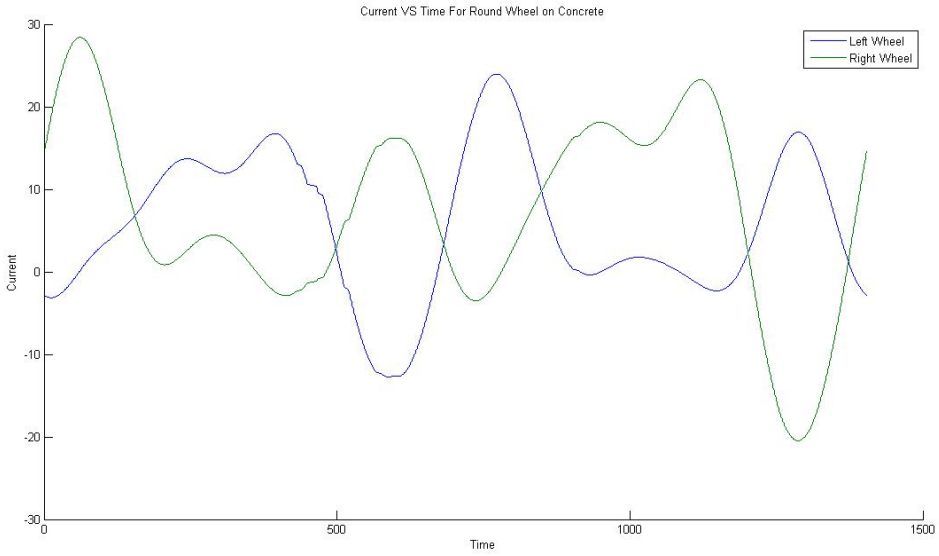


Figure B.4: Round Wheel Current on Concrete.

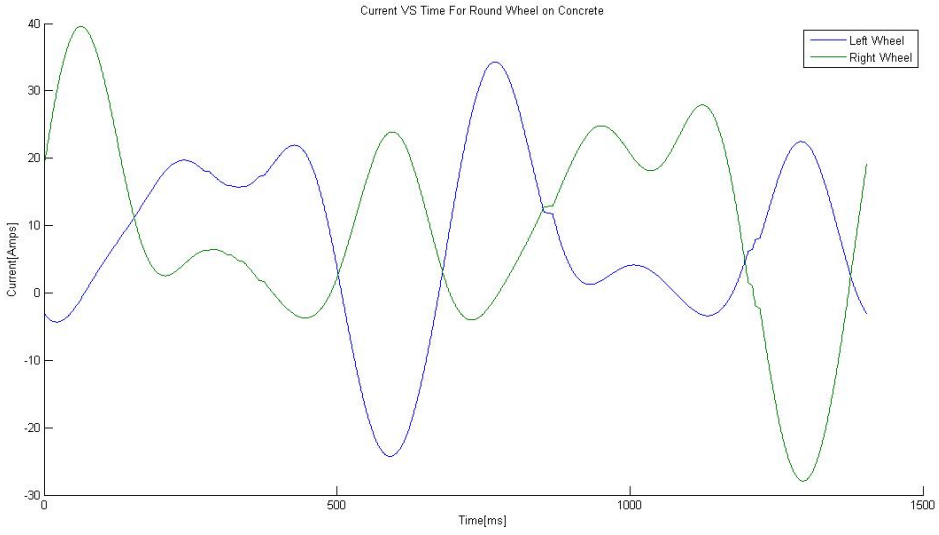


Figure B.5: Rubber Wheel Current on Concrete.

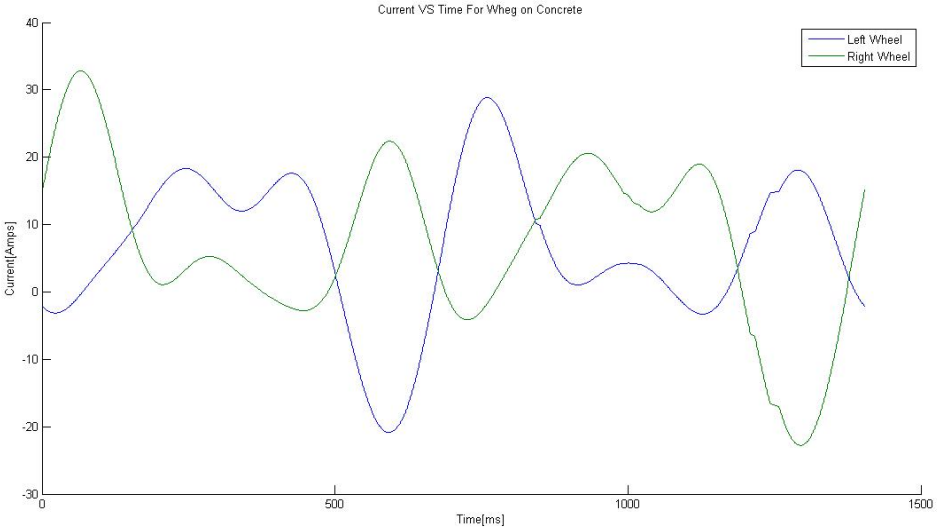


Figure B.6: Wheg Current on Concrete.

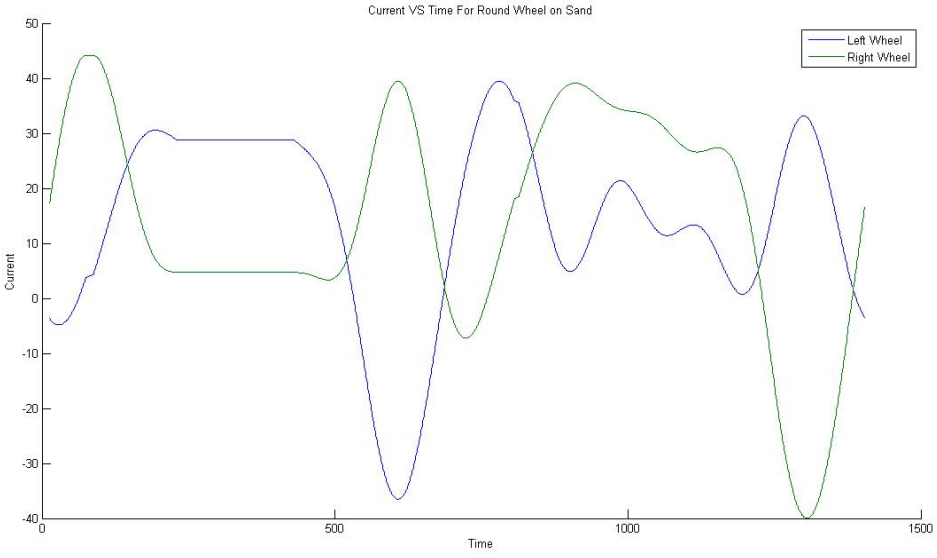


Figure B.7: Round Wheel Current on Sand.

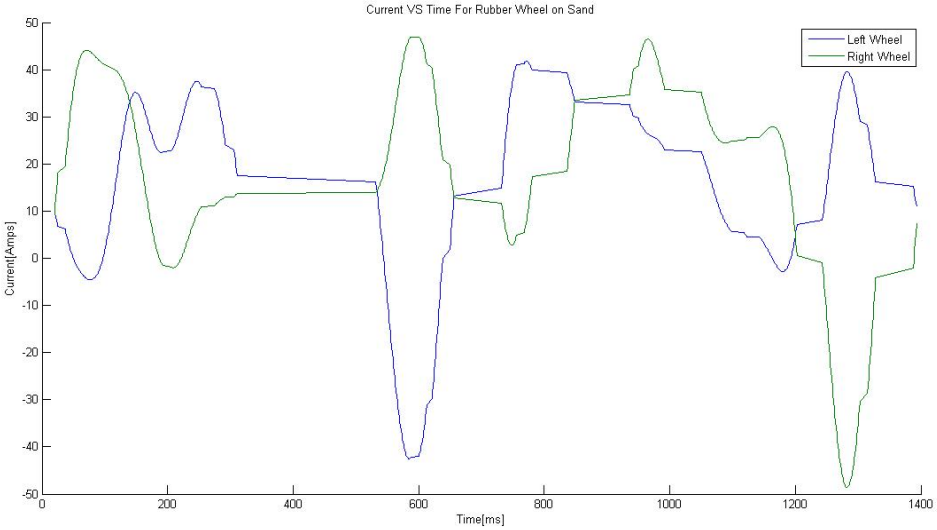


Figure B.8: Rubber Wheel Current on Sand.

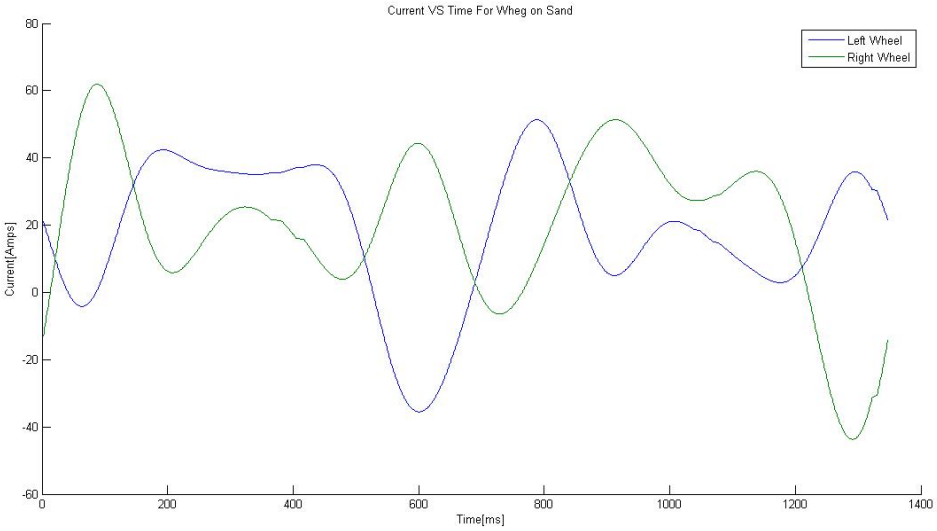


Figure B.9: Wheg Current on Sand.

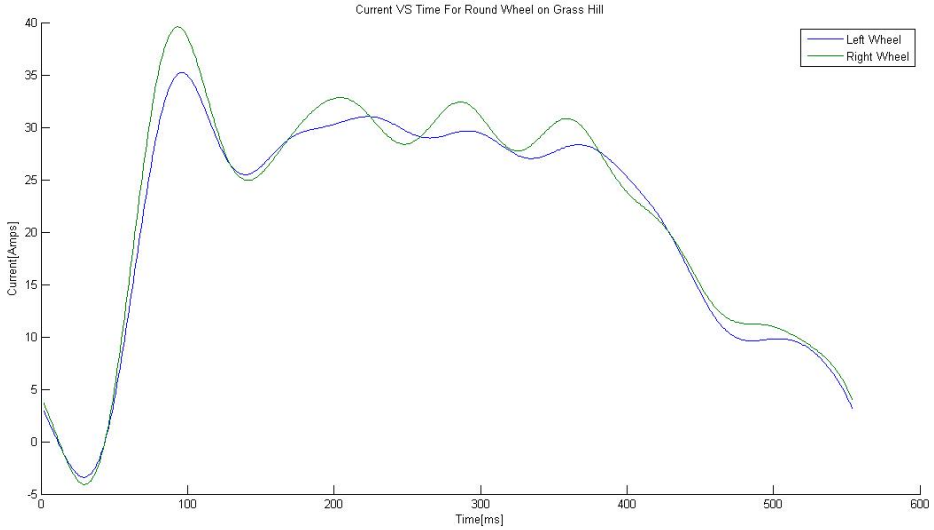


Figure B.10: Rubber Wheel Current on Grass Hill.

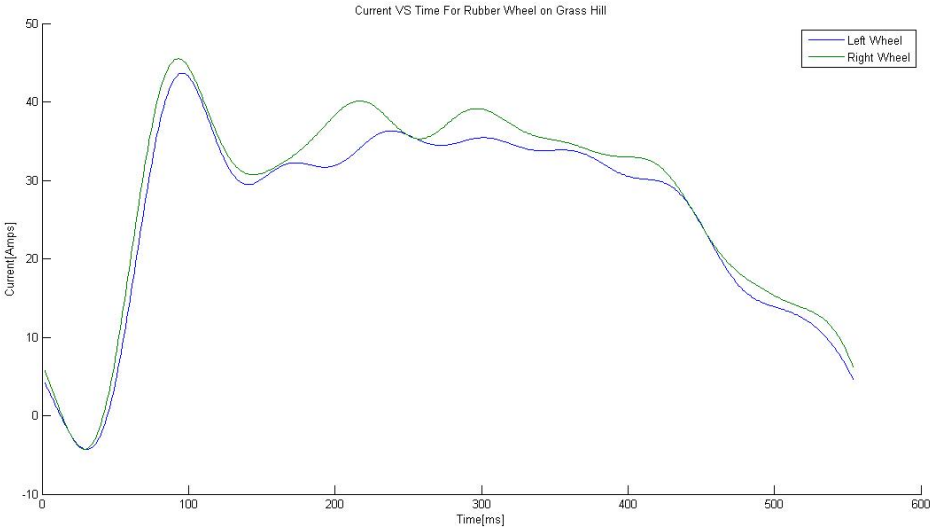


Figure B.11: Rubber Wheel Current on Grass Hill.

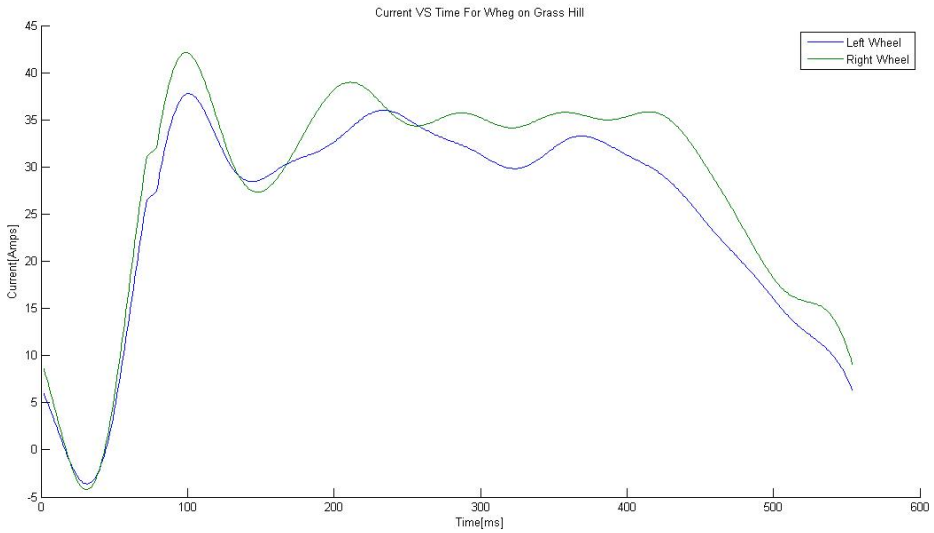


Figure B.12: Wheg Current on Grass Hill.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C:
Raw Current Data

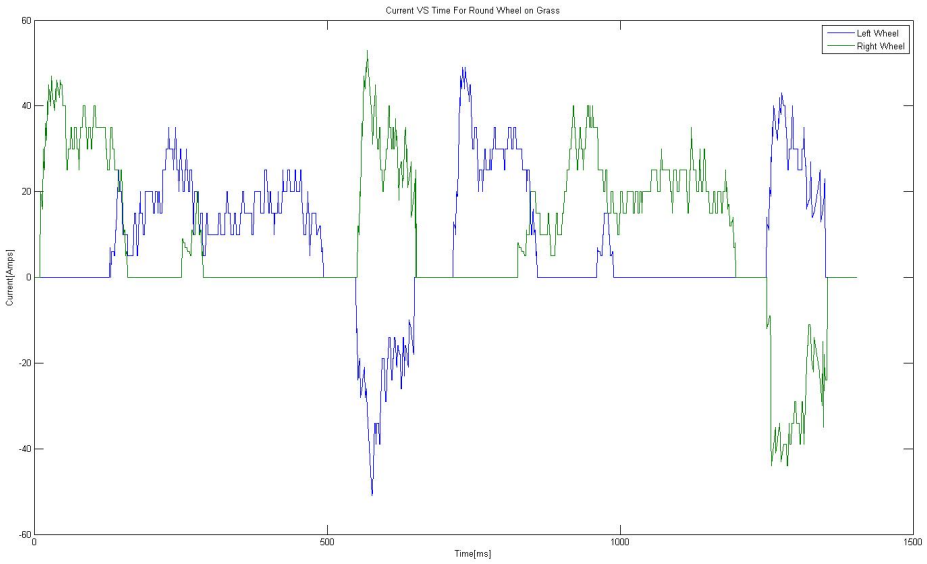


Figure C.1: Round Wheel Current on Grass Raw Data.

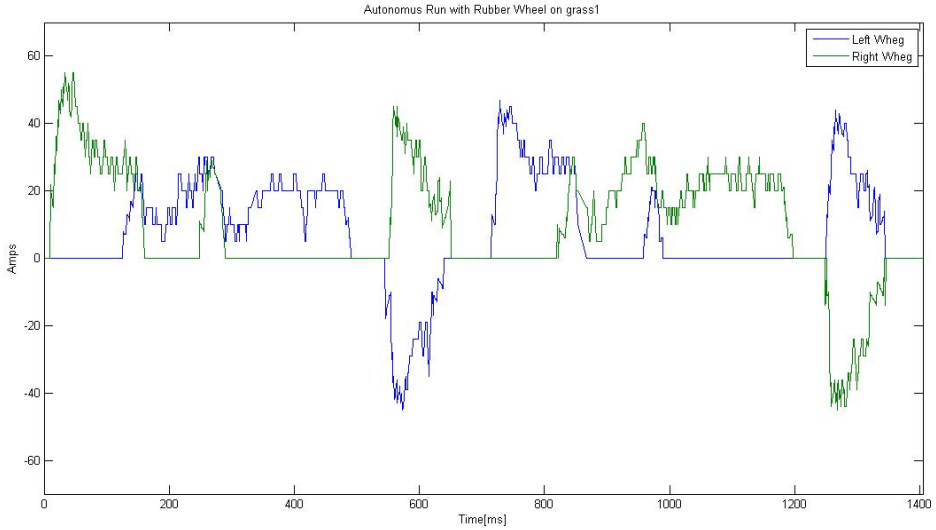


Figure C.2: Rubber Wheel Current on Grass.

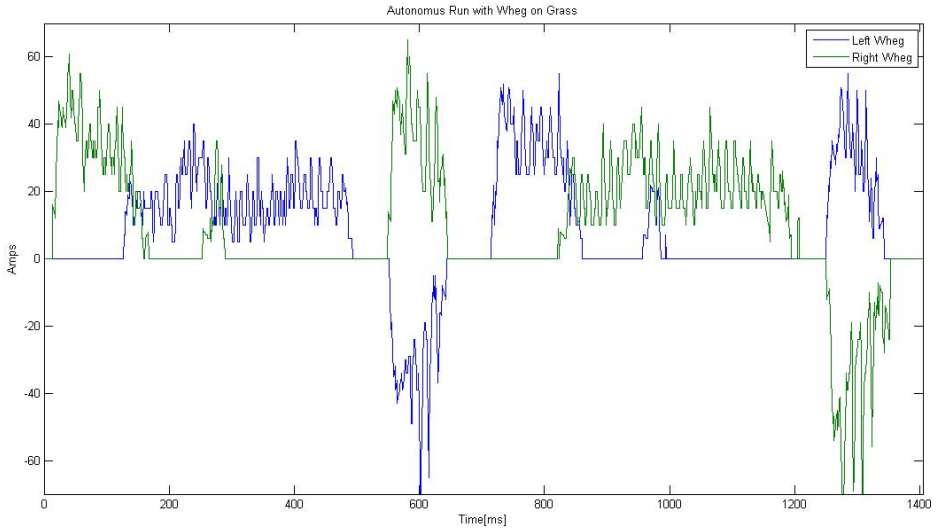


Figure C.3: Wheg Current on Grass.

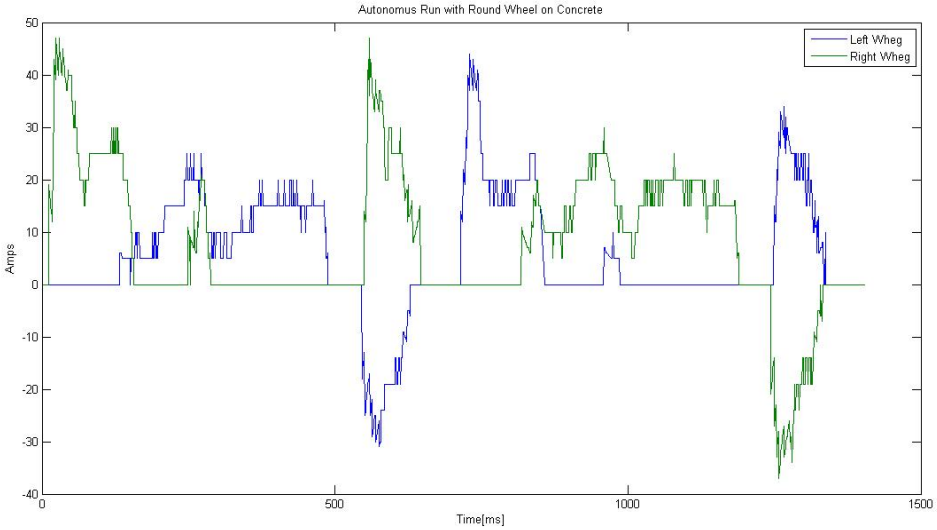


Figure C.4: Raw Round Wheel Current on Concrete.

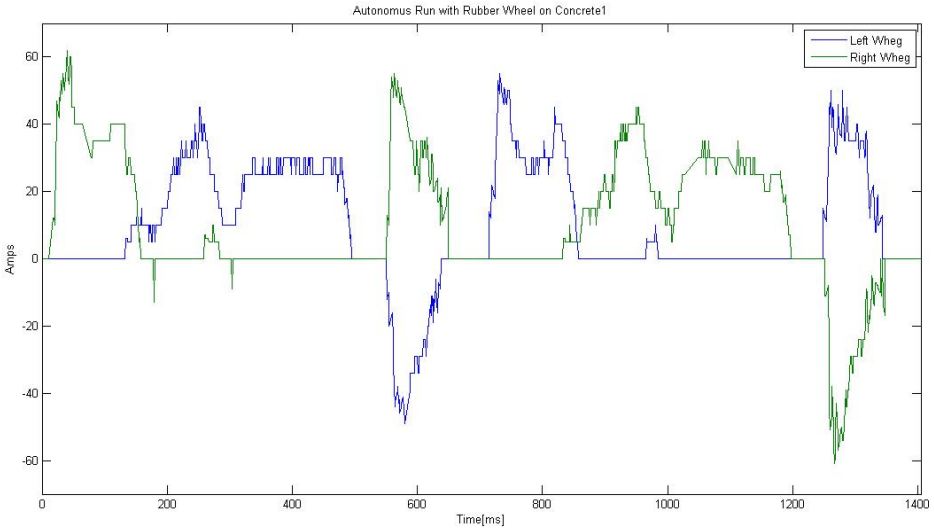


Figure C.5: Raw Rubber Wheel Current on Concrete.

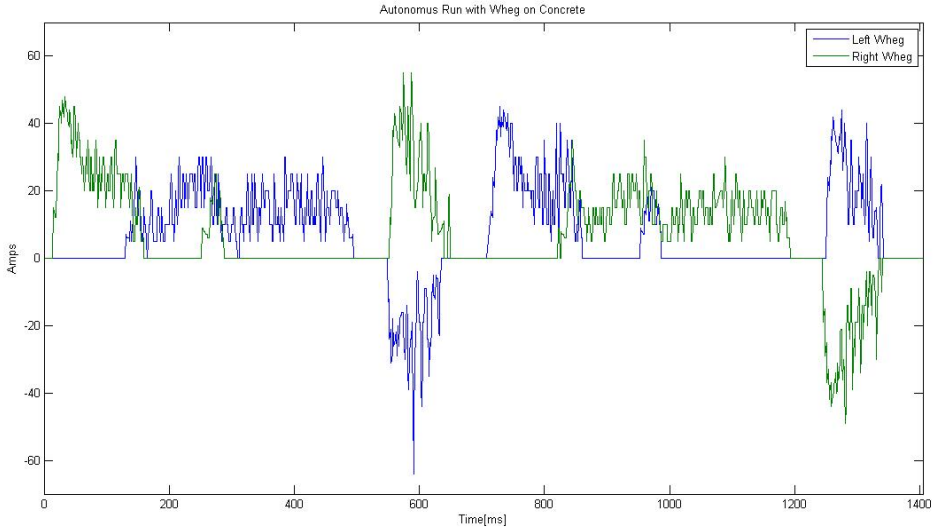


Figure C.6: Raw Wheg Current on Concrete.

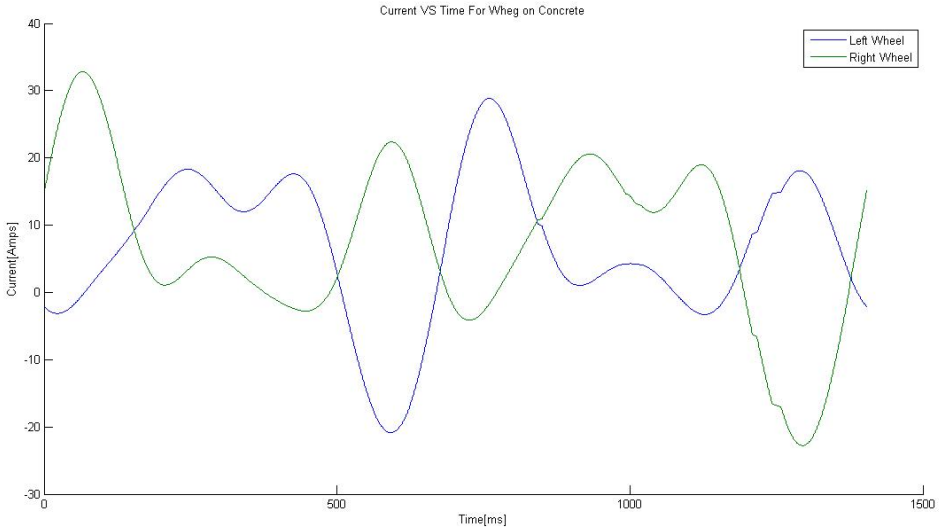


Figure C.7: Wheg Current on Concrete.

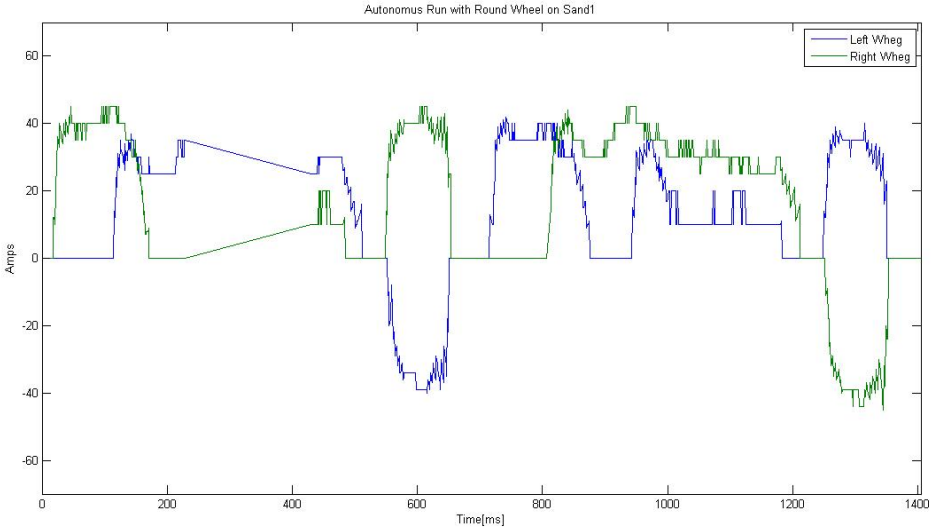


Figure C.8: Raw Round Wheel Current on Sand.

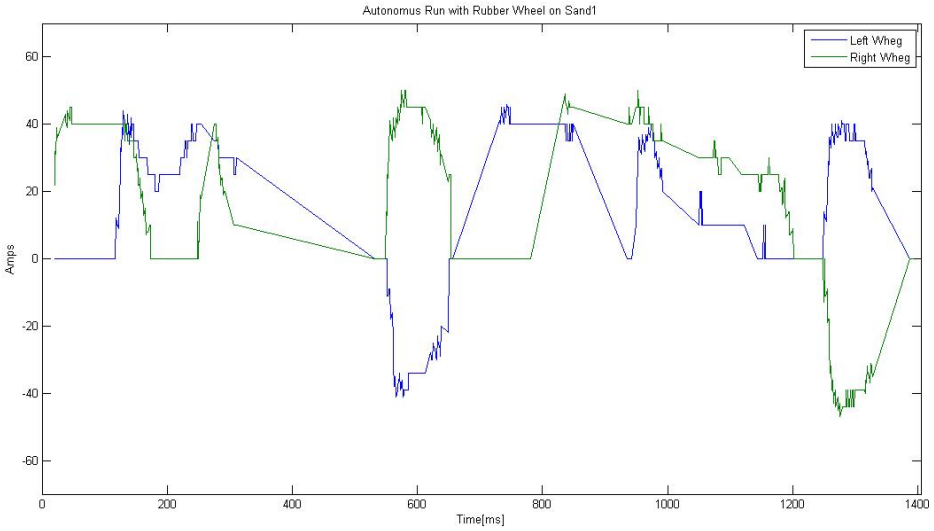


Figure C.9: Raw Rubber Wheel Current on Sand.

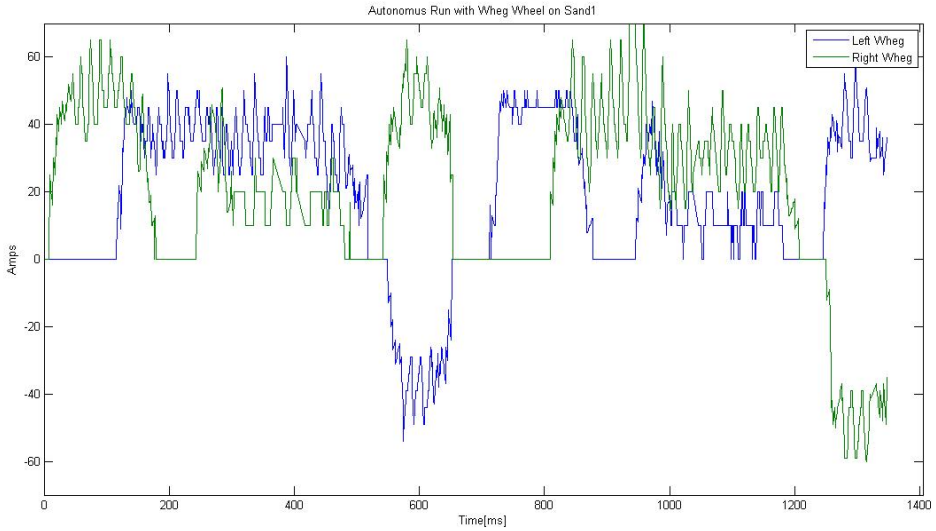


Figure C.10: Raw Wheg Current on Sand.

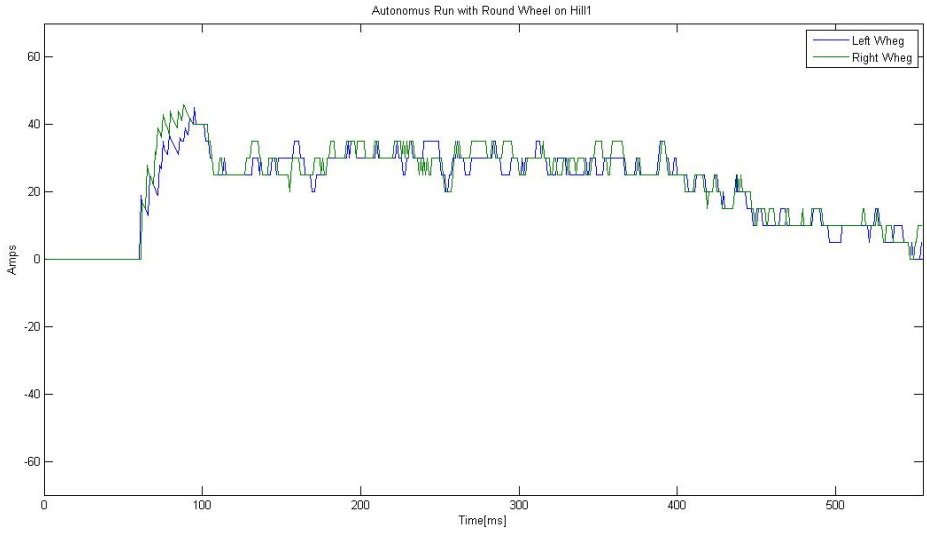


Figure C.11: Raw Round Wheel Current on Grass Hill.

0

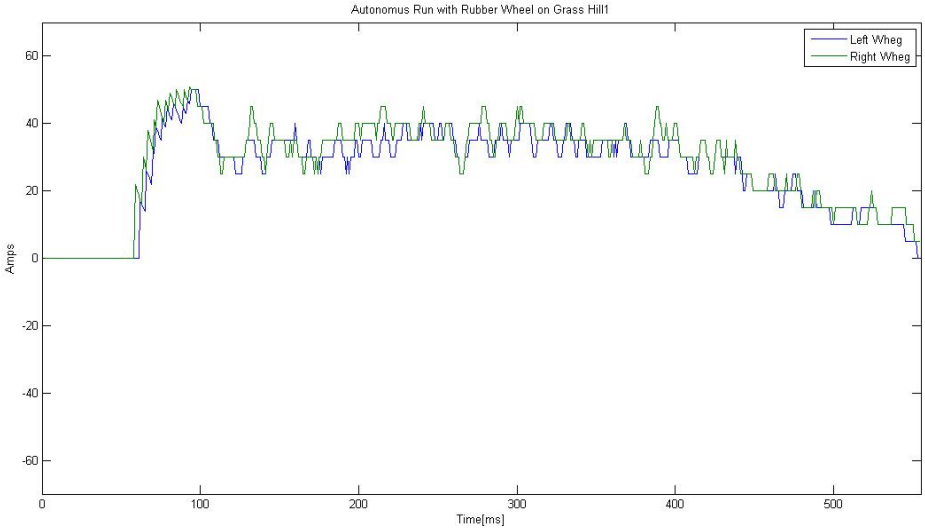


Figure C.12: Raw Rubber Wheel Current on Grass Hill.

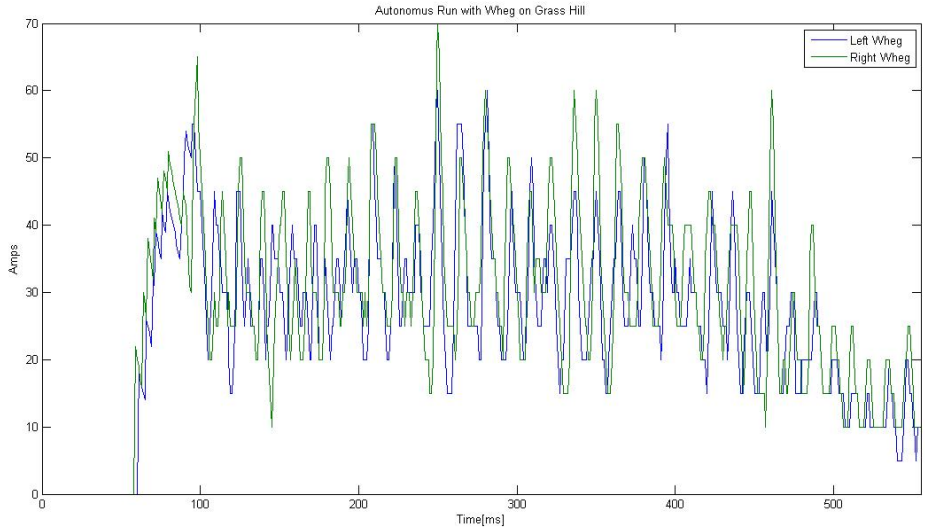


Figure C.13: Raw Wheg Current on Grass Hill.

THIS PAGE INTENTIONALLY LEFT BLANK

List of References

[1] Boston Dynamics, Peichunline. [Online image]. Available: http://peichunlin.me.ntu.edu.tw/Homepage/research/rrhex.htm

[2] T. Ferry, "The NPS small robotic technology initiative, man-portable robots for low intensity conflict," M.S. thesis, Naval Postgraduate School. Monterey, California, 2001.

[3] T. Dunbar, "Demonstration of waypoint navigation for a semi-autonomous prototype Surf-Zone robot," M.S. thesis, Naval Postgraduate School. Monterey, California, 2006.

[4] E. Shuey and M. Shuey, "Modeling and simulation for a surf zone robot," M.S. thesis, Naval Postgraduate School. Monterey, California, 2012.

[5] R. Siegwart et al. "Locomotion," in *Introduction to Autonomous Mobile Robots*, 2nd ed. Cambridge, Massachusetts: The MIT Press, 2004, ch. 2, sec. 2.1.1, pp. 13-14.

[6] J. Fitzgerald, "Characterization parameters for a three degree of freedom mobile robot," M.S., Naval Postgraduate School. Monterey, California, 2013.

[7] IG-42, IG-42 24 VDC Motor. [Online image]. Available: http://www.superdroidrobots.com/shop/item.aspx/ig42-24vdc-122-rpm-gear-motor/490/

[8] Roboteq^{TM}, Roboteq^{TM} 2450. [Online image]. Available: http://www.roboteq.com/index.php/roboteq-products-and-services/brushed-dc-motor-controllers/hdc2450-detail

[9] Roboteq^{TM}, Roboteq^{TM} Navigation. [Online image]. Available: http://www.roboteq.com/index.php/roboteq-products-and-services/navigation-computers

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California